

## **REMARKS**

### **Status of the Claims**

- Claims 1-35 are pending in the Application after entry of this amendment.
- Claims 1-35 are rejected by the Examiner.

### **Interview**

Applicant thanks the Examiner for granting a telephone interview held on February 14, 2007. During that interview, the Applicant's representative explained that one focus of the invention is to perform a consistency check of a database backup using a database view but not actually performing a recovery action. Applicant's representative contrasted the consistency check against the cited art which teaches, among other things, performing a recovery of a primary database. The Examiner cordially acknowledged the difference and requested citations to the specification which teach the use of the database view. Applicant's representative includes that information in this response.

### **Amendment to the Specification**

Applicant amends the specification at paragraphs 0001 and 0002 to include information inadvertently omitted from those paragraphs. The reference information is also in the as-filed specification at paragraphs 0025 and 0026. Applicant submits that no new matter is added to the Specification as a result of including references to related patents and applications.

### **Database View Term in Specification**

Applicant provides the following to facilitate understanding of the term "database view".

Claim 1 of the current application recites a method for performing a consistency check on a database backup which includes, among other elements, creating a database view and recovering the database view to a transaction consistent state. The functionality of the

database view is described in the as-filed specification in many places. As an example, the database view is introduced at paragraph 0026 of the as-filed specification:

“An advancement in the viewing of databases in a transaction consistent manner without creating a complete copy of the database is disclosed in co-pending Patent Application Serial No. 10/611,774 filed June 30, 2003 entitled “Transaction Consistent Copy-On-Write Databases”. This commonly assigned issued patent discloses a storage space efficient method of providing a database view and is hereby incorporated by reference in its entirety. A database view provides a transaction-consistent view of a database at a previous point in time without creating a complete copy of the database.” (paragraph 0026)

Paragraph 0027 of the as-filed specification teaches:

“In an aspect of the present invention, a database view is used as a tool to produce a view of the metadata needed to check the consistency of a database. The database view is also used to store the pages that need the application of redo and undo operations from the transaction log to make the pages transactionally consistent. Such a database view, in combination with the database backup, contains all the information necessary to produce a copy of the original database at the time the full database backup operation completed. The database view, however, does not in itself contain all of the information, and thus can be smaller in size than a full copy of the original database.” (paragraph 0027)

In one embodiment, the database view is preferably used as the environment where the metadata, schema or catalog pages are conveniently located and is the environment where the pages needing redo or undo operations performed from the transaction log are located (see paragraph 0029). A database view is depicted as item 320 in Figure 3 (See paragraph 0033). An example of a transaction consistent database view is depicted as item 480 in Figure 4 and as described in paragraphs 0036 to 0042.

**Claim Rejections Pursuant to 35 U.S.C. §102**

Claims 1-35 stand rejected pursuant to 35 U.S.C. §103(a) as being anticipated by U.S. Patent Publication No. 2003/0061537 to Cha et al. (Cha). Applicant respectfully traverses the rejection.

Cha teaches a parallel logging and recovery scheme for highly available main-memory database systems is presented. A preferred embodiment called parallel redo-only logging ("PROL") combines physical logging and selective replay of redo-only log records (See Cha, Abstract).

Cha performs the parallel logging and recovery using multiple log disks and backup copies of the entire database. (see Cha, paragraphs 0035 and 0036). The process of making entire database backups is termed "checkpointing". As stated in Cha:

"Checkpointing helps throw away the old portion of the log file and thus shorten the log replay time." (paragraph 0007).

"The checkpoint manager 107 is responsible for checkpointing, the process of making backup copies of the entire database from time to time. In the present invention, checkpointing may be done in parallel by partitioning each backup copy into multiple backup disks such as 111 and 112. A preferred system may maintain two backup copies based on the ping-pong checkpointing scheme. The locations of backup databases and important log records such as ones recording the beginning and the end of checkpointing are kept in a log anchor 113. In case of a system failure, a consistent state of the primary database 100 is recovered from the log and the most recent backup database." (paragraph 0036)

Thus, Applicant concludes that the process of "checkpointing" as described by Cha is a process of producing an up to date backup copy of a database that in a manner that "throws away" old log files. Applicant concludes that the more often checkpointing is performed in Cha, the smaller the transaction log file because the backup database that was checkpointed is brought up to date. This has the advantage in Cha of shortening the log replay time (because the transaction log of a checkpointed backup database is shorter than a non-checkpointed backup database transaction log) and this shortened transaction log is useful when a full recovery is performed of the primary database.

However, Applicant cannot find in Cha where a “consistency check” as recited in Claim 1 is performed on the Cha backup database using a “database view” as in the claimed invention. Moreover, Applicant cannot find in Cha where a “database view” requires less memory space than the database contained in the database backup as is recited in Claim 1. Applicant also cannot find where Cha clearly describes a “database view”.

The Office Action dated 12/01/07 on page 2 cites Cha paragraphs 0037 and 0086 to teach the Claim 1 element of “creating a database view from the database backup; wherein the database view requires less memory space than the database contained in the database backup” (Office Action, page 2).

Chu at paragraph 0037 teaches:

“The primary database 100 preferably consists of a number of fixed-size segments, where a segment is the basic unit of memory allocation. A segment is divided into fixed-size pages, and a page is again divided into fixed-size slots. Although the size of a slot is fixed within a page, it may vary by pages. A record is basically stored in one slot while a variable-length record is handled by linking multiple fixed-size slots. The size of a slot is determined when formatting a page in order to allocate it to a table. When a page becomes free by deletion of a record, it is added to the global free page list. The preferred embodiment uses fine-granular, slot-level locking and logging.” (Chu, paragraph 0037)

“There may be a situation where two log records cover either the same portion of the database or non-overlapping portions of the database. FIG. 3 shows such a situation where situation shown in FIG. 3, where the portions of database covered by the log records L1 and L2 overlap each other. Suppose that L1 is encountered after L2 has been replayed. Then, L1 is discarded because it has a smaller update sequence number than L2, and this may result in the loss of information recorded in the shaded area of L1.” (Chu, paragraph 0086)

Respectfully, Applicant can find no teaching in the cited reference at paragraphs 0037 and 0086 that teaches the Claim 1 element of “creating a database view from the database backup; wherein the database view requires less memory space than the database contained in

the database backup”. Chu, paragraph 0037 teaches the separation of memory into segments and the separation of segments into fixed sized page of memory and the allocation of records to slots in a table. Nothing is mentioned concerning database views and the characteristic that the view requires less space than the database contained in a database backup.

Paragraph 0086 of Chu discusses a situation where multiple log records may have some of the same transactions. Since Chu uses multiple log records as part of the method of parallel redo-only logging (PROL) and recovery for databases in main memory, it is logical that Chu discusses this topic. However, Applicant can find little reference in Chu paragraph 0086 concerning “database views” and the characteristic that the database view requires less space than the database contained in a database backup as in Claim 1.

The Office Action on page 2 indicates that Chu paragraphs 0109 and 0208 teach the three Claim 1 elements of “copying a transaction log and the retrieved pages into the database view; recovering the database view to a transaction consistent state; and performing a consistency check on the database backup by reading all database pages from one of the database backup and the database view.” Yet, in studying paragraphs 0109 and 0208, Applicant cannot find any mention of (1) a database view, (2) recovering the database view itself to a transaction consistent state, (3) and the performance of a consistency check by reading pages from the database view or the database backup as recited in Claim 1. In paragraph 0109, Chu discusses algorithms concerning the update, commit, and transaction abort in the slot sequence number (SSN-based) parallel redo-only logging (PROL) as well as log record flushing. No mention is made of a database view or recovery within a database view to a transaction consistent state or the performance of a consistency check as those elements are recited in Claim 1. Chu paragraph 0208 discusses two types of transactions which insert or delete records from a table as well as the overhead for a log table. No mention is made of a database view or recovery within a database view to a transaction consistent state or the performance of a consistency check as those elements are recited in Claim 1.

Since Chu fails to disclose at the least the elements of:

- (i) a database view,
- (ii) a database view that requires less memory space than the database contained in the database backup,
- (iii) recovering the database view to a transaction consistent state,

(iv) a consistency check of a database backup, and  
(v) performing a consistency check on the database backup by reading all database pages from one of the database backup and the database view,

then, Chu cannot anticipate Claim 1 under 35 U.S.C. §102(b) per MPEP §2131 because to anticipate a claim, the reference must teach every element of the claim. Chu fails to fully teach all of the elements of Claim 1.

Applicant notes that independent Claims 17 and 28 also include the structural and functional aspects of a database view as does independent Claim 1. Thus, Claims 1, 17, and 28, along with their respective dependent claims like are not anticipated by Chu.

Applicant thus respectfully requests that the 35 U.S.C. §102(b) rejection of Claims 1-35 be withdrawn because these claims patentably define over the cited art.

**Conclusion**

Applicant respectfully submits that all pending claims patentably define over the cited art. Applicants respectfully request reconsideration and withdrawal of the rejections. A Notice of Allowance for all pending claims is requested.

Respectfully submitted,

Date: March 1, 2007

/Jerome G. Schaefer/

---

Jerome G. Schaefer  
Registration No. 50,800

Woodcock Washburn LLP  
Cira Centre  
2929 Arch Street, 12th Floor  
Philadelphia, PA 19104-2891  
Telephone: (215) 568-3100  
Facsimile: (215) 568-3439